vvddrr: A Fixed-Supply cRank Alternative

6d5916ad93fcd77a74acdacdebe42d2331796a8073c1682516fe0f55187df709
vvddrr.com

November 11, 2022

Abstract

We observe a noted interest in the minting of Xen Crypto. However, we also observe a rapidly increasing total supply without an upper bounds or maximum. Importantly, upon reviewing the Xen Crypto white paper and contract, we observe that there is the presentation of an APY which simply does the following: burns a fixed amount of tokens provided by a user and then returns to re-mint the same amount of tokens with the addition of a percentage above the original amount. This creation of a purely inflationary APY adds no value to the system while further increasing the total amount of available tokens. As an alternative, we can modify and utilize the effective cRank algorithm presented by Xen Crypto while contextualizing its use within a fixed supply of 21,000,000 tokens as a reference to the original Bitcoin maximum supply. Pairing this strategy with the consistent average block time present in Ethereum's shift to Proof of Stake, we can structure staking mechanisms which grow toward a fixed supply while implementing random burn variations. The result is a cRank alternative with fixed-supply and gradual staking emissions.

1 Introduction

The cRank algorithm as implemented by Xen Crypto has value especially in its use of logarithms to value the delta of activity between the point at which a person claims a cRank and that at which they return to mint it relative to the number of claim interactions which have occurred in between. Further innovation is found in allowing a user to set their claim time and thus directly determine their placement within the logarithmic claim curve. However, without a supply restriction, the Xen Crypto cRank algorithm has the effect of creating ever larger total claimed amounts. In a general sense, use of such large numerical amounts are unnecessary in any ERC20 or similar environment as it is common place to use 18 decimal places to provide units of value. Instead of an increasing supply or large mint values, we can structure token based rewards using decimal values and a fixed maximum supply.

To understand the systematic risks of an ever increasing monetary supply and thus a token for which there is no maximum supply, we only need to observe the current global inflation rates. In the crypto ecosystem specifically, the creation of increased token supply or alternative reward token supply as a method to provide APY can also be observed to have resulted in detrimental effects. To provide a more direct example, assume we are placing 100,000 USD into an interest bearing savings account at a rate of 2% for a period of a year. If the bank were to simply return our 100,000 USD at the end of the year and then print our 2% interest–thereby increasing the overall monetary supply–our original 100,000 USD is thereby less valuable at the end of the year. The bank has provided no value.

However, were the bank concurrently to lend 100,000 USD to a home developer who is willing to pay a 5% interest rate on it since they aim to get a 10% return on the home they are building, the bank has created value and my 2% return is a real APY. The bank, within a closed system and using a fixed supply, has utilized market dynamics to create a transfer of value. While Bitcoin does not engage in the active creation of value as does a bank, it utilizes a fixed supply emitted over a multi-year horizon to avoid the effects of an unrestricted monetary supply. Bitcoin can also be thought of as a human-variation network. Human variation resulting in missing Bitcoin keys, previously mined Bitcoin that remains untouched, and simply forgotten or lost Bitcoin results in a total circulation supply that is less than the final maximum.

2 Fixed Supply cRank Modification

Shifting the cRank algorithm to a fixed supply model using the ERC20 decimal depth is achieved using the following algorithm modification:

$$R_u = 0.38055175038 \cdot \left(1 + \frac{\log_2(cR_g - cR_u)}{100}\right) \cdot \frac{T_u}{T_m} \cdot (1 + EAA(cR_u)) \qquad (1)$$

While leaving the current maximum term calculation methodology unchanged:

$$T_m = \begin{cases} 100, if cR_g \leq 5000 \\ 100 + \log_2(cRg) \cdot 15, if cR_u > 5000 \end{cases} \qquad (2)$$

And making the following modification to the EAA equation:

$$EAA(cR_u) = 0.1 - 0.001 \cdot \left[\frac{cR_u}{2600000}\right] \qquad (3)$$

Given that:

$$cR_u \leq cR_g \qquad (4)$$

And:

$$cR_g \leq 2682000 \qquad (5)$$

We begin to calculate the reward received by a user by utilizing the shift by Ethereum to Proof of Stake which gives a stable 12 second block time. Translated into a yearly value gives a total of 2,628,000 blocks per year. Distributing a baseline value of 1,000,000 million vvddrr tokens per year gives a rate of 380517503800000000 wei per block or 0.3805175038 vvddrr per block. We then use this baseline as the initial decimal value to calculate a user's reward, cR_u, rather than an ever increasing count of tokens and set a maximum total claims amount of 2,628,000 for cR_g.

We keep the $\log_2(\mathrm{cR}_g - cR_u)$ value as we can determine that the log of our highest input value 2,627,999 will be 21.3255333. We convert the log value to a percentage gain over 100% so that the maximum log value increase would approximate 121%. We then evaluate the term that a user chooses to set their mint term, T_u, as a percentage of the maximum available term, T_m. Thus users who set a very short term will receive a significantly less percentage of the total available reward for a given mint instance. We apply this as a direct percentage increase or decrease.

We keep the EAA rate calculation but we set the 100,000 denominator to 2,600,000. This allows the reward to drop every 300,000 cRanks. We then enforce conditions that require the current cR_u to be below the maximum allowable global rank, cR_g. From the original equation, we remove the reward amplifier value, $AMP(ts_0)$, as we utilize decimal depth to provide reward value within a fixed-supply rather than a large mint quantity.

Additionally, for each mint reward claim, we also calculate a value between 0 and the base reward claim value to burn. This burn amount is not withdrawn from a user's reward claim but is minted and burned separately. The burn amount is generated using a keccak256 hash of a phrase, the block timestamp, and stringified values of a user's cR_u, cR_g, and T_u. This automated burn mechanism also adds a deflationary element to vvddrr should users choose predominantly to implement cRank mints as a way of minting vvddrr. The total mount of vvddrr burned through this process will not exceed 1,000,000.

As a general example, if you have a claim cRank, cR_u, of 1, set your term to the full 100 of the max term available, and have a global cRank, cR_g, of approximately 1,200,000 when you return after 100 days to mint your vvddrr, we would take the base value in wei of 380517503800000000, multiply it by 120% to represent your log value of 20 over 100, multiply again by 100% for your selection of the full term available, and then by 110% since your EAA rate is the highest possible. We then divide by a percentage basis denominator and you have approximately 5 vvddrr tokens that you earned by getting an early cRank and holding for the maximum 100 days while 1,200,000 cRanks were claimed to increase the globalRank.

Even if all 2,628,000 cRank claims were set to maximize their terms and are withdrawn only after that maximum is reached, the total vvddrr minted would be below 13,310,000. Another contrasting method of minting the remaining vvddrr is thus implemented which relies on a more gradual emissions process. Implementing the mechanism for vvddrr to reach the 21,000,000 total supply also utilizes the Ethereum Proof of Stake shift to a 12 second block time. Setting an emission value of 0.0000038051750381 or 3805175038100 wei per block gives 100,000 users 10 tokens per year for a total of 1,000,000 tokens emitted yearly. This value can then be interpreted into three distinct ERC721 mining rig tokens with varying percentage ranges within sets of 100,000. Utilizing IERC721 staking, vvddrr can be emitted gradually over a period of years with a distribution method the exact dynamics of which are reliant on randomly allocated percentage values.

3 Mining Rig Percentage Calculations

vvddrr uses a controller pattern allowing three staking contracts to mint and burn vvddrr within the restrictions of the fixed supply maximum. There is a staking contract for each mining rig type: fixed, variable and micro. Each of these staking contracts has the same per block reward rate set as a constant: 3805175038100 wei. Variation in received rewards is derived from the percentage value allocated to each type of mining rig at mint. This value is stored in a fixed array and does not change. While fixed mining rigs have a 100 percent value, variable mining rigs have a percentage value between 50 and 100 while micro have a percentage value between 10 and 50. This value is calculated at mint using the combination of the following parameters: a kecca256 hash of a phrase, the msgSender address, the block timestamp value, and a string of the current token id. A user's received mining rewards for a given staking period are then multiplied by this mining rig percentage at the time of withdrawal.

4 Mining Rig Maximum Supply

Each type of mining rig (fixed, variable, and micro) is limited to 100,000 units. This constraint allows for the calculation of upper and lower bounds for each rig as all of them have the same 3805175038100 tokens per block reward value hard-coded into their staking contracts. Using the 12 second Etheruem PoS block time gives 2,628,000 blocks per year and thus a total of 10 vvddrr tokens per year. Each fixed vvddrr mining rig will thus have an equal upper and lower bounds of 10 vvddrr tokens per year. For 100,000, that is then 1 million vvddrr tokens minted per year before burn capitalization minting given full purchase and staking.

As that the percentage rewards value of a variable mining rig can be between 50 percent and 150 percent, given that all are minted at the lower value or the upper value, we have either 5 vvddrr tokens or 15 vvddrr tokens per year in rewards, respectively. For 100,000, that gives either 500,000 vvddrr tokens or 1.5 million vvddrr tokens per year. As the value of a mmccrr mining rig can be between 10 percent and 50 percent, that is between 1 and 5 vvddrr tokens per year. For 100,000, that is between 100,000 and 500,000 vvddrr tokens per year emitted as rewards. This setting of mining rig maximum supply based on types gives an expected reward amount for each mining rig type that will result in the gradual emission of vvddrr over a number of years while adding variation due to the random allocation of percentages within a range at mint.

5 Mining Rig Burn Capitalization

Each mining rig regardless of type has a default burn capitalization value that is set at 3805175038100 wei when the mining rig is first staked. We define the burn capitalization as a uint256 value between zero and which a value shall be chosen to be burned on each unstake action. The cumulative value of the reward allocated to the unstake action and the calculated burn capitalization derived value must always be below 21 million. When a percentage payout is claimed during unstake, that uint256 value is then set as the updated burn capitalization between zero and which the new burn amount for the subsequent unstake will be calculated.

Some examples can illustrate the implementation of a burn capitalization. If you have a fixed mining rig and stake it for the first time until it has accumulated 1 vvddrr in rewards, when you unstake to claim your rewards, your burn capitalization is at 3805175038100 wei. The staking contract will then calculate a value between 0 and 3805175038100 to add as the amount of vvddrr to be burned from the total available supply of vvddrr that can be minted in order to process your unstake. So you will receive 1 vvddrr and say 67805175038100 wei of vvddrr will also be burned. You thus receive your vvddrr and some vvddrr from the total available 21 million vvddrr is created and burned. The burn capitalization for your mining rig is now set at 1 vvvddrr or 1000000000000000000 wei meaning on your next unstake, a value between 0 and 1000000000000000000 will be selected to be burned and set as the burn capitalization value.

The deflationary effects of mining rig burn capitalization are thus directly correlated to human action and the ways in which people choose to interact with the vvddrr staking mechanisms. If people predominantly do micro stakes and repeatedly stake and unstake, there will be overall low burn capitalizations and low burned amounts. If more people stake their mining rigs for longer periods and unstake less frequently, the chance that there will be larger and larger overall burn capitalizations increases, leading to more vvddrr being created but also burned. (Note, however, that the burn actions will be less frequent in this case). If someone, for examples, allows 10 vvddrr to accumulate before unstaking then there is possibility that their burn capitalization could fall between 0 and 10000000000000000000 wei. Human choices thus directly influence what can be referred to as the cumulative vvddrr staking burn rate. Additionally, as the maximum supply of 21,000,000 created vvddrr approaches after a few years of staking, anyone who has accumulated a very high amount of rewards as well as a high burn capitalization will have to consider the viability of claiming their rewards with the addition of a randomized burn calculation. Collective human actions will result in varying outcomes.

6 Boolean Set Restrictions

In addition to not utilizing pause mechanisms, the vvddrr contract ecosystem implements boolean set restrictions whenever a value is required to be set after the point of contract initialization. As the default state of a boolean in Solidity is false, a false equality check before function execution with a change to a true state as part of function execution without any other state changes available locks in any set values. In the vvddrr ERC20 contract, this includes the setting of the initial genesis timestamp as well as the setting of the fixed, variable and micro staking controllers. In each staking contract, the setting of the vvddrr ERC20 address and the relevant ERC721 token address also use boolean set restrictions.

7 Conclusion

We observed a significant interest in minting cRank tokens but we also observed a large supply of tokens minted with the current implementation and no maximum supply as is a central part of Bitcoin's value creation. By modifying the cRank algorithm to incorporate a fixed supply and adding a fixed per block reward value, we have created an ERC20 token that allows for the rapid interactions of cRank claims and the more gradual emissions of ERC721 staking. While blockchain utilization allows for extensive contract based minting, rapid minters with short terms receive greatly reduced rewards. Human variance in unstake and therefore burn capitalization decision times along with a burn at each instance of a cRank mint claim adds a deflationary pressure to vvddrr similar to the real-world variance of lost or unmoved Bitcoin.